Presented by: Malak Mohamed Abd El-Moneim,

Nadeen Ahmed Nabil

Presented to: Eng/Sara

Id:220591 ,220594

HealthCare Analytics Department

Bookset Assignment

1.

IMPORTING THE DATASET

```
[5]: import pandas as pd
     import matplotlib.pyplot as plt

[6]: # Load the dataset
     df = pd.read_csv("C:/Users/HP/Desktop/books 5.csv")
```

- Firstly, we imported packages as pandas and matplotlib as pandas allows you to read and write data from various file formats, clean and preprocess data, perform data aggregation and summarization and matplotlib provides a wide range of plotting functions and styles to create various types of plots, including line plots, scatter plots, bar plots, histograms
- Then we used the pd.read_csv() function from the pandas library to read a CSV file located at the given path. It loads the data from the CSV file into a dataframe object named df, allowing for data manipulation and analysis in Python.

2.

```
[7]: df.head()
```

[7]:

	book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors	original_publication_year	original_title	...	ratings_count
0	1	2767052	2767052	2792775	272	439023483	9.780439e+12	Suzanne Collins	2008.0	The Hunger Games	...	4780653
1	2	3	3	4640799	491	439554934	9.780440e+12	J.K. Rowling, Mary GrandPré	1997.0	Harry Potter and the Philosopher's Stone	...	4602479
2	3	41865	41865	3212258	226	316015849	9.780316e+12	Stephenie Meyer	2005.0	Twilight	...	3866839
3	6	11870085	11870085	16827462	226	525478817	9.780525e+12	John Green	2012.0	The Fault in Our Stars	...	2346404
4	12	13335037	13335037	13155899	210	62024035	9.780062e+12	Veronica Roth	2011.0	Divergent	...	1903563

- This function is used to display the first few rows of the df. By default, it shows the first 5 rows, providing a quick overview of the data structure and content.

3.

DATA CLEANING

```
[8]: # REMOVING NULLS
     print("Missing Values:")
     print(df.isnull().sum())
     df.dropna(inplace=True)
     print(df.isnull().sum())
```

```
Missing Values:
book_id                     0
goodreads_book_id           0
best_book_id                0
work_id                     0
books_count                 0
isbn                       52
isbn13                     44
authors                     0
original_publication_year   3
original_title             52
title                       0
language_code             109
average_rating              0
ratings_count               0
work_ratings_count          0
work_text_reviews_count     0
ratings_1                   0
ratings_2                   0
ratings_3                   0
ratings_4                   0
ratings_5                   0
image_url                   0
```

- The code first prints the number of missing values in each column then, it drops rows with any missing values. Finally, it prints the updated count of missing values in each column.

4.

```
[9]: # Drop unnecessary columns
     # df = df.drop(['goodreads_book_id', 'best_book_id', 'work_id', 'isbn', 'isbn13', 'language_code', 'image_url', 'small_image_url'], axis=1)
```

```
[10]: df.head()
```

[10]:

	book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors	original_publication_year	original_title	...	ratings_count	wo
0	1	2767052	2767052	2792775	272	439023483	9.780439e+12	Suzanne Collins	2008.0	The Hunger Games	...	4780653	
1	2	3	3	4640799	491	439554934	9.780440e+12	J.K. Rowling, Mary GrandPré	1997.0	Harry Potter and the Philosopher's Stone	...	4602479	
2	3	41865	41865	3212258	226	316015849	9.780316e+12	Stephenie Meyer	2005.0	Twilight	...	3866839	
3	6	11870085	11870085	16827462	226	525478817	9.780525e+12	John Green	2012.0	The Fault in Our Stars	...	2346404	
4	12	13335037	13335037	13155899	210	62024035	9.780062e+12	Veronica Roth	2011.0	Divergent	...	1903563	

5 rows × 23 columns

- It removes specific columns from the df and after executing this code, the df will no longer contain these columns.

5.

```
[11]: # Remove duplicates if any
      df = df.drop_duplicates()
```

```
[12]: # Convert original_publication_year to integer and handle missing values
      df['original_publication_year'] = df['original_publication_year'].fillna(0).astype(int)
```

DATA PREPROCESSING

```
[13]: # Filter dataset for Harry Potter series
      hp_df = df[df['title'].str.contains('Harry Potter')]
```

```
[14]: hp_df.head()
```

[14]:

	book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors	original_publication_year	original_title	...	ratings_count	w
1	2	3	3	4640799	491	439554934	9.780440e+12	J.K. Rowling, Mary GrandPré	1997	Harry Potter and the Philosopher's Stone	...	4602479	
6	18	5	5	2402163	376	043965548X	9.780440e+12	J.K. Rowling, Mary GrandPré, Rufus Beck	1999	Harry Potter and the Prisoner of Azkaban	...	1832823	
								J.K. Rowling		Harry Potter and the			

- We make sure to remove any duplicates.
- This line of code fills missing values in the 'original_publication_year' column with 0 and the column is represented as integer values.
- It creates a new df containing rows where the 'title' column contains the substring 'Harry Potter' so it filters the rows based on this condition.

6.

```
[16]: # Convert necessary columns to appropriate data types
hp_df['books_count'] = hp_df['books_count'].astype(int)
hp_df['average_rating'] = hp_df['average_rating'].astype(float)
hp_df['ratings_count'] = hp_df['ratings_count'].astype(int)
hp_df['work_ratings_count'] = hp_df['work_ratings_count'].astype(int)
```

- These lines of code convert the data types of specific columns and these conversions ensure that the data in these columns is represented in the appropriate numeric data types for further analysis or visualization.

7.

```
[17]: # Find most selling Harry Potter books
most_selling_books = hp_df.sort_values(by='books_count', ascending=False)

# Print most selling Harry Potter books
print("Most Selling Harry Potter Books:")
print("------------------------------")
for index, row in most_selling_books.iterrows():
    print(f"Title: {row['title']}")
    print(f"Books Sold: {row['books_count']}")
    print("------------------------------")
```

```
Most Selling Harry Potter Books:
------------------------------
Title: Harry Potter and the Sorcerer's Stone (Harry Potter, #1)
Books Sold: 491
------------------------------
Title: Harry Potter and the Chamber of Secrets (Harry Potter, #2)
Books Sold: 398
------------------------------
Title: Harry Potter and the Prisoner of Azkaban (Harry Potter, #3)
Books Sold: 376
------------------------------
Title: Harry Potter and the Goblet of Fire (Harry Potter, #4)
Books Sold: 332
------------------------------
Title: Harry Potter and the Order of the Phoenix (Harry Potter, #5)
Books Sold: 307
------------------------------
Title: Harry Potter and the Half-Blood Prince (Harry Potter, #6)
Books Sold: 275
```

- This code sorts the df by the 'books_count' column in descending order to find the most selling Harry Potter books. Then, it prints the titles of these books along with the number of books sold.

8.

```
[21]: # Calculate average rating of Harry Potter books
average_rating = hp_df['average_rating'].mean()
```

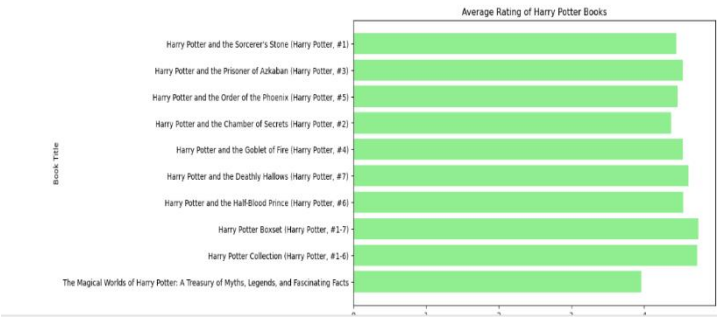
```
[19]: # Print average rating of Harry Potter books
print("Average Rating of Harry Potter Books:")
print("-------------------------------------")
print(f"Average Rating: {average_rating:.2f}")
print("-------------------------------------")
```

```
Average Rating of Harry Potter Books:
-------------------------------------
Average Rating: 4.49
-------------------------------------
```

- It accesses the 'average_rating' column from the df and then calculates the mean of the ratings . The resulting value is stored in the variable average_rating.
- It first prints a header indicating that it's displaying the average rating of Harry Potter books. Then, it prints the calculated average rating with two decimal places using f-strings

9.

```
[20]: # Plotting the average rating of Harry Potter books
plt.figure(figsize=(10, 6))
plt.barh(hp_df['title'], hp_df['average_rating'], color='lightgreen')
plt.xlabel('Average Rating')
plt.ylabel('Book Title')
plt.title('Average Rating of Harry Potter Books')
plt.gca().invert_yaxis()
plt.show()
```



- This code snippet creates a horizontal bar plot to where each bar represents the average rating of a Harry Potter book, and the titles of the books are shown on the y-axis.